```
1       /*
          File:     dv_spis2.cpp

          Compiler: BC++ 3.1

          Задача.   Сформировать двусвязный список из N элементов.

                    Примечание.
                      Для создания списка использовать переменную типа
                      указатель на указатель, для создания массива
                      указателей  на элементы списка.

                    Знакомство с типом данных УКАЗАТЕЛЬ на УКАЗАТЕЛЬ.

          Решение.  MSP,
                    CNIT "North Star",
                    02.11.07, 12:53, 17:14
        */

14      # include <stdio.h>
15      # include <conio.h>

16      // структура одного элемента двусвязного списка
17      class element
18      {
19        public:
20          element * L; // указатель на предыдущий элемент списка
21          int data; // ключ элемента списка
22          element * R; // указатель на следующий элемент списка
23      };


24      int main ()
25      {
26        clrscr ();

27        int N; // длина двусвязного списка

28        printf ("Сколько элементов в списке чисел: ");
29        scanf ("%i", & N);
30        fflush (stdin);

31        // Под каждый элемент списка нужно будет выделить 10 байт.
32        // Нам придется N раз выделять память и тут же прописывать
33        // значения левого и правого указателей.
34        // Процесс непростой, нужно все рисовать на бумаге.

35        int i; // номер элемента двусвязного списка

36        element * * A; // A - это указатель типа element *
```

```
37                      // или, как говорят программисты,
38                      // это УКАЗАТЕЛЬ на УКАЗАТЕЛЬ: * *

39      // спросим у OS адрес, по которому она расположит массив из
40      // N адресов, причем каждый из этих адресов есть адрес какого-то
41      // элемента списка
42      A = new element * [N]; // память под N адресов

43      // теперь у нас есть место для хранения N адресов,
44      // но пока неизвестны САМИ адреса элементов.
45      // Сделаем это.

46      // выделим память под хранение всех элементов списка
47      for (i = 0; i < N; i ++)
48        A [i] = new element;

49      // теперь все адреса ликвидны, правильны, точно заданы самой OS,
50      // осталось по этим адресам записать информацию по всем полям
51      // элемента списка
52      for (i = 0; i < N; i ++)
53        A [i] -> data = i;

54      // Запишем NULL в самый левый указатель и в самый правый указатель
55      A [0] -> L = NULL;
56      A [N - 1] -> R = NULL;

57      // пропишем все ПРАВЫЕ указатели
58      for (i = 0; i < N - 1; i ++)
59        A [i] -> R = A [i + 1];

60      // пропишем все ЛЕВЫЕ указатели
61      for (i = 1; i < N; i ++)
62        A [i] -> L = A [i - 1];

63      // двусвязный список создан, все указатели заполнены

64      printf ("печать списка в обратном порядке: \n");

65      // напечатаем список в обратном порядке
66      element * tek;

67      tek = A [N - 1];
68      while (tek != NULL)
69      {
70        printf ("%i ", tek -> data);
71        tek = tek -> L;
72      }

73      // Важнейщий этап.
74      // Возврат израсходованной памяти OS.
75      // Если запрос памяти производили таким образом, что
```

```
76        // сначала попросили память под массив указателей,
77        // и только потом, под каждый элемент списка,
78        // то ВОЗВРАТ памяти OS будем производить в ОБРАТНОМ порядке:
79        // сначала вернем память, занятую элементами списка,
80        // а потом вернем память, занятую массивом указателей на
81        // элементы списка.

82        // 1. Вернем память, занятую элементами
83        for (i = 0; i < N; i ++)
84          delete A [i];

85       // 2. Вернем память, занятую массивом указателей
86       delete [] A;

87        // А теперь вопрос.
88        // Можно ли без дополнительной переменной A
89        // (указатель на указатель)
90        // решить задачу создания двусвязного списка?
91        //  -- Можно.

92        getch ();

93        return 0;
94      }
```

Listing данной задачи опубликован в сети Internet по адресу
http://www.Best-Listing.ru/color-3-task-13.html

Sergey Mitrofanov, 30.10.13, 20:55
E-mail: infostar @ mail.ru